UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
09/849,754	05/04/2001	G. Glenn Henry	CNTR:2062	2670

23669 7590 03/18/2004

HUFFMAN LAW GROUP, P.C.
1832 N. CASCADE AVE.
COLORADO SPRINGS, CO 80907-7449

EXAMINER	
GERSTL, SHANE F	
ART UNIT	**PAPER NUMBER**
2183	4

DATE MAILED: 03/18/2004

Please find below and/or attached an Office communication concerning this application or proceeding.

Office Action Summary	Application No. 09/849,754	Applicant(s) HENRY ET AL.	
	Examiner Shane F Gerstl	Art Unit 2183	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If the period for reply specified above is less than thirty (30) days, a reply within the statutory minimum of thirty (30) days will be considered timely.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on 18 November 2002 and 18 June 2002.

2a)☐ This action is **FINAL**. 2b)☒ This action is non-final.

3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) 1-38 is/are pending in the application.

4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)☐ Claim(s) _____ is/are allowed.

6)☒ Claim(s) 1-38 is/are rejected.

7)☐ Claim(s) _____ is/are objected to.

8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)☒ The specification is objected to by the Examiner.

10)☒ The drawing(s) filed on 04 May 2001 is/are: a)☒ accepted or b)☐ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)☐ All b)☐ Some * c)☐ None of:

1.☐ Certified copies of the priority documents have been received.

2.☐ Certified copies of the priority documents have been received in Application No. _____.

3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) ☒ Notice of References Cited (PTO-892)

2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)

3) ☒ Information Disclosure Statement(s) (PTO-1449 or PTO/SB/08) Paper No(s)/Mail Date 2.

4) ☐ Interview Summary (PTO-413) Paper No(s)/Mail Date. _____ .

5) ☐ Notice of Informal Patent Application (PTO-152)

6) ☐ Other: _____.

DETAILED ACTION

1. Claims 1-38 have been examined.

Papers Received

2. Receipt is acknowledged of information disclosure statement and change of address papers submitted, where the papers have been placed of record in the file.

Specification

3. The disclosure is objected to because of the following informalities: The serial numbers of the copending applications of page 1 are blank and need to be filled in. Also, the last copending application with docket number CNTR: 2063 is not in records of the US Patent Office. If this application no longer exists, the reference to it should be removed.

4. The lengthy specification has not been checked to the extent necessary to determine the presence of all possible minor errors. Applicant's cooperation is requested in correcting any errors of which applicant may become aware in the specification.

5. The disclosure is objected to because of the following informalities: The section titled "Summary" should be titled Summary of the Invention as shown in 37 CFR 1.77(b).

6. The disclosure is objected to because of the following informalities: The summary should be directed to the invention which is set forth in the claims. The summary of this disclosure, however, is identical to the summaries of the copending applications and

thus is not directed to the claimed invention of this application but to all the inventions. See MPEP 608.01(d).

Appropriate correction is required.

Claim Objections

7. Claim 1 is objected to because of the following informalities: lines 8-9 of the second paragraph recite the limitation "said associated branch instruction". It has only been stated previously that the plurality of offsets are associated with a plurality of previously executed branch instructions and not that each offset is associated with each previously executed branch instruction. The examiner is taking lines 4-7 to mean, "said plurality of cached target addresses and offsets each associated with one of a plurality of previously executed branch instructions" as implied by the claim language and stated in the specification. Also, the examiner requests that the limitation "said associated branch instruction be rephrased to say, "said associated previously executed branch instruction," so that the claim language is consistent and clear.

Appropriate correction is required.

Claim Rejections - 35 USC § 112

8. The following is a quotation of the second paragraph of 35 U.S.C. 112:

> The specification shall conclude with one or more claims particularly pointing out and distinctly claiming the subject matter which the applicant regards as his invention.

9. Claims 2 and 14-18 are rejected under 35 U.S.C. 112, second paragraph, as being indefinite for failing to particularly point out and distinctly claim the subject matter which applicant regards as the invention.

10. Claim 2 recites the limitation "said one of said plurality of branch instructions" in line 1 of page 114. There is insufficient antecedent basis for this limitation in the claim. Only a plurality of previously executed branch instructions has been defined. The examiner is taking the limitation to mean said associated branch instruction in order to fit the context of the parent claim.

11. Claim 14 states, "...one of said plurality of target addresses that has a smallest said corresponding one of said plurality of offsets." There is insufficient antecedent basis for this limitation in the claim. There is no "said corresponding one of said plurality of offsets" defined. The examiner is taking this limitation to mean "one of said corresponding plurality of offsets" as is stated in the parent claim 13.

Claim Rejections - 35 USC § 102

12. The following is a quotation of the appropriate paragraphs of 35 U.S.C. 102 that form the basis for the rejections under this section made in this Office action:

> A person shall be entitled to a patent unless –
>
> (b) the invention was patented or described in a printed publication in this or a foreign country or in public use or on sale in this country, more than one year prior to the date of application for patent in the United States.

13. Claims 1-7, 10-16, and 18-36 are rejected under 35 U.S.C. 102(b) as being anticipated by Shiell (5,850,543).

14. In regard to claim 1, Shiell discloses a pipelined microprocessor, comprising:

a. an instruction cache, configured to receive a fetch address on an address bus; Figure 1 shows an instruction cache 16 on an address bus that extends to the level 2 cache 14 and fetch unit 26. Figure 2 shows the fetch unit 26 and that

it sends a PA (physical address) derived from an FA (fetch address) to the instruction cache on the instruction bus.

b. a branch target address cache (BTAC), coupled to said address bus, configured to provide a plurality of cached target addresses and offsets in response to said fetch address, said plurality of cached target addresses and offsets associated with a plurality of previously executed branch instructions, each of said plurality of offsets specifying a location of said associated branch instruction within a line of said instruction cache; Figure 2, element 56 and column 7, lines 39-45 show that a branch target buffer that stores target addresses in a cache arrangement (and is hence a branch target address cache) is in the fetch unit for generating subsequent fetch addresses (FA). Figure 2 and column 8, line 12 show that the instruction bus is coupled to the BTAC. Column 8, lines 34-46 show that the cache stores a tag that includes the target address and an offset that indicate the address of a previously performed branching instruction. Lines 36-43 particularly show that the offset is used for addressing the specific instruction (in this case the branch) within the code line (of the cache since the fetch unit retrieves instructions from the cache as shown in figure 2) associated with the logical address (fetch address). Column 11, lines 50-57 show that if the BTAC (BTB) does not contain a tag (the target address and offset) because the branch (a CALL in this case) has not yet been executed, upon execution of the branch the BTAC is updated with the tag. Column 13, lines 16-23 show that the tag (and thus the target address and offset) is provided

in response to encountering the branch instruction again (and thus is previously executed).

c. and branch control logic, coupled to said BTAC, for generating a selector signal in response to said fetch address and said plurality of offsets, said selector signal selecting one of said plurality of target addresses provided by said BTAC as a subsequent fetch address on said address bus. As shown above, the BTAC generates subsequent fetch addresses and branch instruction fetch addresses are matched with cache entries to obtain these fetch addresses. This matching inherently provides a select signal that chooses the appropriate target address and is sent for fetching as shown in figure 2. It is also inherent that there is logic needed for this matching and selection and since it controls the branch target address cache an appropriate name would be branch control logic.

15. In regard to claim 2, Shiell discloses the microprocessor of claim 1, wherein said selected one of said plurality of target addresses is selected as said subsequent fetch address regardless of whether said one of said plurality of branch instructions associated with said selected one of said plurality of target addresses is present in a line of instructions in said instruction cache that is selected by said fetch address. As shown figure 2, the BTAC is in the fetch unit for speculative execution as shown in the column 2, lines 38-63. Because the BTAC is in the fetch unit, the type of instruction is not yet known for the fetch address because it has not yet been decoded. It is only presumed (speculated) that the instruction at the fetch address is again a branch if found in the BTAC. Column 10, lines 55-58 show that the invention jumps to other code

segments of memory. In such an event, the fetch address will point to a different instruction than the branch even if one is selected in the BTAC due to the fetch address. The disclosure gives no indication that the BTAC will not select a target address every time a match is found in the BTAC regardless of the code segment. Thus regardless of if a branch instruction is present in the cache line, a target address is selected.

16. In regard to claim 3, Shiell discloses the microprocessor of claim 1, wherein said branch control logic generates said selector signal to only select said one of said plurality of target addresses whose associated offset is greater than or equal to a portion of said fetch address. As shown above, the tag matches the BTAC with a branch instruction. Also as shown above, the tag, and more specifically the offset, includes the fetch address to match. Since these addresses are matched, the target address is selected according to the associated offset, which is equal to a portion of the fetch address.

17. In regard to claim 4, Shiell discloses the microprocessor of claim 3, wherein said portion of said fetch address comprises a plurality of least significant bits of said fetch address. As shown above and in column 8, lines 34-46 the tag contains the address of the branch instruction. Also as shown above, the offset is the portion of the fetch address for matching. Figure 3 shows that the tag or logical address (LA) is seamless with the offset, which is part of this address. The figure also shows that the offset is the least significant bits of this fetch address.

18. In regard to claim 5, Shiell discloses the microprocessor of claim 4, wherein said plurality of least significant bits of said fetch address specify a byte offset of said

associated branch instruction within a line of said instruction cache selected by said fetch address. As shown above, the offset is a least significant portion of the fetch address of an associated branch instruction. In the table of column 6, predecode stage 1 shows that the instructions of use (including the associated branch instruction) are instruction bytes. Thus, the offset that indicates the fetch address of these instruction bytes, is appropriately named a byte offset.

19. In regard to claim 6, Shiell discloses the microprocessor of claim 4, wherein said plurality of least significant bits of said fetch address comprises a number of bits corresponding to a number of bits comprising said offset. As shown above, the offset comprises the least significant bits of the fetch address.

20. In regard to claim 7, Shiell discloses the microprocessor of claim 3, wherein if a plurality of said plurality of offsets is greater than or equal to said portion of said fetch address, said selector signal is used to select said one of said plurality of target addresses whose associated offset is a smallest of said plurality of said plurality of offsets greater than or equal to said portion of the fetch address. The use of alternative language warrants the examiner to take the case where there is a plurality of offsets equal to said portion of said fetch address. As shown above, the control logic generates said selector to select said one of said plurality of target addresses. With the offsets that are equal to the portion of the fetch address available for selection of a target address that corresponds to them, the selector will inherently select the smallest one which is any of the offsets.

21. In regard to claim 10, Shiell discloses the microprocessor of claim 1, wherein said BTAC is configured to provide a direction prediction associated with each of said plurality of branch instructions for predicting whether said associated branch instruction will be taken or not taken. Column 8, line 47 – column 9, line 20 show that a history field is included in each line of the BTAC that gives a prediction on whether an associated branch will be taken or not.

22. In regard to claim 11, Shiell discloses the microprocessor of claim 10, wherein said selector signal is used to select said one of said plurality of target addresses only if said associated direction prediction predicts that said associated branch instruction will be taken. It is inherent that if the branch is predicted as not taken that the sequential address is then used for the next instruction instead of the branch target address. Thus the selector signal is only used when the branch will be taken.

23. In regard to claim 12, Shiell discloses the microprocessor of claim 1, further comprising: address selection logic, coupled to said BTAC, for selecting said one of said plurality of target addresses as said subsequent fetch address in response to said selector signal. As shown above, when there is a hit in the branch target address cache, a branch target address is selected for subsequent fetching. This is inherently done using some sort of address selection logic.

24. In regard to claim 13, Shiell discloses an apparatus for selecting a target address for one of a plurality of previously executed branch instructions, said plurality of previously executed branch instructions being potentially present in a line of an

instruction cache (figure 1, element 16) selected by a fetch address, the fetch address provided to the instruction cache on an address bus, the apparatus comprising:

a. a branch target address cache (BTAC), coupled to the address bus, configured to provide a plurality of target addresses cached therein in response to the fetch address, and to provide a corresponding plurality of offsets within the instruction cache line for each of the plurality of previously executed branch instructions; Figure 2, element 56 and column 7, lines 39-45 show that a branch target buffer that stores target addresses in a cache arrangement (and is hence a branch target address cache) is in the fetch unit for generating subsequent fetch addresses (FA). Column 8, lines 34-46 show that the cache stores a tag that includes the target address and an offset that indicate the address of a previously performed branching instruction. Lines 36-43 particularly show that the offset is used for addressing the specific instruction (in this case the branch) within the code line (of the cache since the fetch unit retrieves instructions from the cache as shown in figure 2) associated with the logical address (fetch address). Column 11, lines 50-57 show that if the BTAC (BTB) does not contain a tag (the target address and offset) because the branch (a CALL in this case) has not yet been executed, upon execution of the branch the BTAC is updated with the tag. Column 13, lines 16-23 show that the tag (and thus the target address and offset) is provided in response to encountering the branch instruction again (and thus is previously executed).

b. control logic, coupled to said BTAC, for generating a selector in response

to the fetch address and said offsets, said selector for selecting one of said plurality of target addresses; As shown above, the BTAC generates subsequent fetch addresses and branch instruction addresses are matched with cache entries to obtain these fetch addresses. This matching inherently provides a select signal that chooses the appropriate target address and is sent for fetching as shown in figure 2. It is also inherent that there is logic needed for this matching and selection and since it controls the branch target address cache an appropriate name would be branch control logic.

c. and address selection logic, coupled to the selector, for selecting one of said plurality of target addresses as a subsequent fetch address for the instruction cache in response to said selector, said address selection logic selecting said one of said plurality of target addresses as said subsequent fetch address for the instruction cache regardless of how many branch instructions are present in the instruction cache line selected by the fetch address. As shown above, an address is selected for fetching and thus address selection logic is inherent. As shown figure 2, the BTAC is in the fetch unit for speculative execution as shown in the column 2, lines 38-63. Because the BTAC is in the fetch unit, the type of instruction is not yet known for the fetch address because it has not yet been decoded. It is only presumed (speculated) that the instruction at the fetch address is again a branch if found in the BTAC. Column 10, lines 55-58 show that the invention jumps to other code segments of memory. In such an event, the fetch address will point to a different instruction than the branch even if

one is selected in the BTAC due to the fetch address. The disclosure gives no indication that the BTAC will not select a target address every time a match is found in the BTAC regardless of the code segment. Thus regardless of if a branch instruction is present in the cache line (or regardless of the number of branch instructions), a target address is selected.

25. In regard to claim 14, Shiell discloses the apparatus of claim 13, wherein said control logic generates said selector to select said one of said plurality of target addresses that has a smallest one of said corresponding plurality of offsets. As shown above, the control logic generates said selector to select said one of said plurality of target addresses. Shiell only gives indication that an offset corresponds to each instruction (as shown previously) in the instruction cache but does not give indication that there is more than one offset for any instruction. Therefore, with only one offset, the selector selects the target address corresponding to that offset, which happens to also be the smallest offset.

26. In regard to claim 15, Shiell discloses the apparatus of claim 14, wherein said smallest corresponding offset is greater than or equal to a corresponding portion of the fetch address. As shown above, the tag matches the BTAC with a branch instruction. Also as shown above, the tag, and more specifically the offset, includes the fetch address to match. Since these addresses are matched, the target address is selected according to the associated offset, which is equal to a portion of the fetch address.

27. In regard to claim 16, Shiell discloses the apparatus of claim 15, wherein said control logic generates said selector to select said one of said plurality of target

addresses having said smallest said corresponding one of said plurality of offsets if said BTAC predicts that one of the plurality of branch instructions corresponding to said smallest said corresponding one of said plurality of offsets will be taken. Column 8, line 47 – column 9, line 20 show that a history field is included in each line of the BTAC that gives a prediction on whether an associated branch will be taken or not. It is inherent that if the branch is predicted as not taken that the sequential address is then used for the next instruction instead of the branch target address. Thus the selector signal is only used when the branch will be taken.

28. In regard to claim 18, Shiell discloses the apparatus of claim 15, wherein said control logic generates said selector to select said one of said plurality of target addresses having said smallest said corresponding one of said plurality of offsets if said BTAC indicates that the fetch address hit in said BTAC. Column 13, lines 13-53 show that when the fetch address is "hit" in the BTAC that the appropriate target address is then selected from the cache.

29. In regard to claim 19, Shiell discloses the apparatus of claim 13, wherein said BTAC provides said plurality of target addresses cached therein for a subset of the instruction cache line. Since the branch instructions are from a line of the instruction cache, they can appropriately be considered a subset of the instruction cache line for which their target address have been already shown to be contained in the BTAC.

30. In regard to claim 20, Shiell discloses the apparatus of claim 13, wherein said BTAC provides said plurality of target addresses prior to decoding of said instruction cache line. As shown above and in figures 1 and 2, the fetch unit contains the BTAC;

the results of which are sent to the predecode units of figure 1. Thus the target address are provided before decoding.

31. In regard to claim 21, Shiell discloses the apparatus of claim 13, wherein the fetch address is a virtual address, wherein said BTAC provides said plurality of target addresses based on the virtual fetch address without converting the virtual fetch address to a physical address. Column 8, lines 36-38 show that the tag of the BTAC stores a logical address or virtual address. Figure 2 and column 5, lines 55-58 show that the logical fetch address (FA) retrieved from the BTAC 56 is converted by a TLB 22 into a PA (physical address) that is output by the multiplexer and thus the virtual and not physical address is stored before conversion.

32. In regard to claim 22, Shiell discloses the apparatus of claim 13, wherein the plurality of previously executed branch instructions potentially present in the instruction cache line comprises a plurality of return instructions, wherein said plurality of offsets provided by said BTAC comprises offsets for said plurality of return instructions. Column 12, line 66- column 13, lines 12 show that the BTB (BTAC) contains return instructions with a tag entry (that includes the offset field).

33. In regard to claim 23, Shiell discloses the apparatus of claim 22, further comprising: a call/return stack, coupled to said BTAC, for providing a return address to said address selection logic. Column 12, lines 28-30 shows that there is a return address stack for return instructions. Figure 2 shows that this stack 55 is coupled to the BTAC 56.

34. In regard to claim 24, Shiell discloses the apparatus of claim 23, wherein said control logic is configured to generate said selector to selectively control said address selection logic to select said return address provided by said call/return stack in response to said plurality of offsets and the fetch address. Since the return instructions are stored in the BTAC just as other branch instructions, the control logic selects the return address in response to the plurality of offsets and the fetch address just as before.

35. In regard to claim 25, Shiell discloses an apparatus for selecting a branch target address in a pipelined microprocessor having an instruction cache, a fetch address provided to the instruction cache on an address bus selecting a line of instructions therein, the apparatus comprising:

a. a branch target address cache (BTAC), coupled to the address bus, for providing information cached therein about a plurality of previously executed branch instructions in response to the fetch address, said information comprising a plurality of target addresses associated with said plurality of previously executed branch instructions; Figure 2, element 56 and column 7, lines 39-45 show that a branch target buffer that stores target addresses in a cache arrangement (and is hence a branch target address cache) is in the fetch unit for generating subsequent fetch addresses (FA). Column 8, lines 34-46 show that the cache stores a tag that includes the target address and an offset that indicate the address of a previously performed branching instruction. Lines 36-43 particularly show that the offset is used for addressing the specific instruction (in

this case the branch) within the code line (of the cache since the fetch unit retrieves instructions from the cache as shown in figure 2) associated with the logical address (fetch address). Column 11, lines 50-57 show that if the BTAC (BTB) does not contain a tag (the target address and offset) because the branch (a CALL in this case) has not yet been executed, upon execution of the branch the BTAC is updated with the tag. Column 13, lines 16-23 show that the tag (and thus the target address and offset) is provided in response to encountering the branch instruction again (and thus is previously executed).

b. and control logic, coupled to said BTAC, for selecting as a subsequent fetch address on the address bus one of said plurality of target addresses associated with one of said plurality of branch instructions, said subsequent fetch address selected in response to said information and the fetch address; As shown above, the BTAC generates subsequent fetch addresses and branch instruction addresses are matched with cache entries to obtain these fetch addresses. This matching inherently provides a select signal that chooses the appropriate target address and is sent for fetching as shown in figure 2. It is also inherent that there is logic needed for this matching and selection and since it controls the branch target address cache an appropriate name would be branch control logic.

c. wherein said control logic selects said one of said plurality of target addresses that is predicted taken and that is first seen with respect to the fetch address, said one of said plurality of target addresses selected whether or not a

branch instruction is present in the line of instructions. Column 8, line 47 – column 9, line 20 show that a history field is included in each line of the BTAC that gives a prediction on whether an associated branch will be taken or not. It is inherent that if the branch is predicted as not taken that the sequential address is then used for the next instruction instead of the branch target address. Thus the selector signal is only used when the branch will be taken.

36. In regard to claim 26, Shiell discloses the apparatus of claim 25, wherein said control logic is configured to generate an indication that one of said plurality of target addresses provided by said BTAC was selected as said subsequent fetch address, wherein said indication is provided to an instruction buffer for receiving the line of instructions. Figure 2 shows an instruction buffer 60 coupled to receive an indication from the BTAC, which has been show to select the subsequent fetch address. Column 7, lines 65-67 show that in response to the physical address (translated from the fetch address for presentation to the cache as shown in figure 2), the instruction cache presents the instruction code (line of instructions) to the instruction buffer.

37. In regard to claim 27, Shiell discloses the apparatus of claim 26, wherein said indication is provided to the instruction buffer for association with one of the instructions in the line of instructions, said one of the instructions presumably corresponding to said one of the plurality of branch instructions that is associated with said selected one of said plurality of target addresses. As shown previously the BTAC presents an indication to the instruction buffer for association with the received line of instructions. This is

presumably a branch instruction since the BTAC contains information for branch instructions.

38. In regard to claim 28, Shiell discloses the apparatus of claim 27, wherein said indication is associated in the instruction buffer with said one of the instructions based on a location within the instruction cache line of said one of the plurality of branch instructions that is associated with said selected one of said plurality of target addresses, said location comprised in said information provided by said BTAC. It is inherent that the information provided by the BTAC for association with the instruction line of the instruction buffer is based on a location in the cache line because there is no other way to know which instructions are associated with the branch targets except with location information such as an address stored in the BTAC.

39. In regard to claim 29, Shiell discloses a method for selecting a fetch address to provide to an instruction cache for speculatively branching a microprocessor, the method comprising:

a. providing a plurality of target addresses and instruction cache line offsets of a corresponding plurality of previously executed branch instructions, in response to a first fetch address provided to the instruction cache; Figure 2, element 56 and column 7, lines 39-45 show that a branch target buffer that stores target addresses in a cache arrangement (and is hence a branch target address cache) is in the fetch unit for generating subsequent fetch addresses (FA). Column 8, lines 34-46 show that the cache stores a tag that includes the target address and an offset that indicate the address of a previously performed

branching instruction. Lines 36-43 particularly show that the offset is used for addressing the specific instruction (in this case the branch) within the code line (of the cache since the fetch unit retrieves instructions from the cache as shown in figure 2) associated with the logical address (fetch address). Column 11, lines 50-57 show that if the BTAC (BTB) does not contain a tag (the target address and offset) because the branch (a CALL in this case) has not yet been executed, upon execution of the branch the BTAC is updated with the tag. Column 13, lines 16-23 show that the tag (and thus the target address and offset) is provided in response to encountering the branch instruction again (and thus is previously executed).

b. determining, based on said plurality of offsets, which of said previously executed branch instructions is located after said first fetch address; As shown above, the BTAC generates subsequent fetch addresses and branch instruction addresses are matched with cache entries to obtain these fetch addresses.

c. and selecting, as a second fetch address to provide to the instruction cache, in response to said determining, one of said plurality of target addresses corresponding to one of said branch instructions that is located after said first fetch address and which is nearest said first fetch address. As shown above one of the target addresses is selected. When there is only one target address match, this is the nearest address to the first fetch address.

40. In regard to claim 30, Shiell discloses the method of claim 29, wherein said selecting comprises selecting said second fetch address whether or not a branch

instruction is present in a line of instructions in the instruction cache selected by said first fetch address. As shown figure 2, the BTAC is in the fetch unit for speculative execution as shown in the column 2, lines 38-63. Because the BTAC is in the fetch unit, the type of instruction is not yet known for the fetch address because it has not yet been decoded. It is only presumed (speculated) that the instruction at the fetch address is again a branch if found in the BTAC. Column 10, lines 55-58 show that the invention jumps to other code segments of memory. In such an event, the fetch address will point to a different instruction than the branch even if one is selected in the BTAC due to the fetch address. The disclosure gives no indication that the BTAC will not select a target address every time a match is found in the BTAC regardless of the code segment. Thus regardless of if a branch instruction is present in the cache line, a target address is selected.

41. In regard to claim 31, Shiell discloses the method of claim 29, further comprising: caching said plurality of target addresses and said instruction cache line offsets of said corresponding plurality of previously executed branch instructions prior to said providing. As shown above, a BTAC is used to cache the target addresses and offsets of the previously executed branch instructions.

42. In regard to claim 32, Shiell discloses the method of claim 29, wherein said first and second fetch addresses are virtual addresses. Column 8, lines 36-38 show that the tag of the BTAC stores a logical address or virtual address. Figure 2 and column 5, lines 55-58 show that the logical fetch address (FA) retrieved from the BTAC 56 is

converted by a TLB 22 into a PA (physical address) that is output by the multiplexer and thus the virtual and not physical address is stored before conversion.

43. In regard to claim 33, Shiell discloses the method of claim 29, wherein said plurality of previously executed branch instructions comprise x86 branch instructions. The table at the bottom of column 11 shows that x-86 type instructions are fetched and decoded and thus the branch instructions are x-86 instructions.

44. In regard to claim 34, Shiell discloses the method of claim 29, wherein said providing said plurality of target addresses comprises providing two target addresses per a subset of a line of instructions that is selected by said first fetch address. There is no mention in the disclosure of Shiell that limits the number of branch instructions in a cache line to one. Thus a subset of a line may occur with multiple branches and thus two or more target addresses would be selected.

45. In regard to claim 35, Shiell discloses the method of claim 29, further comprising: providing a plurality of direction predictions of said corresponding plurality of previously executed branch instructions in response to said first fetch address. Column 8, line 47 – column 9, line 20 show that a history field is included in each line of the BTAC that gives a prediction on whether an associated branch will be taken or not.

46. In regard to claim 36, Shiell discloses the method of claim 35, wherein said selecting comprises selecting as said second fetch address said one of said plurality of target addresses corresponding to one of said branch instructions located after said first fetch address which is nearest said first fetch address and which is predicted taken by a corresponding one said plurality of direction predictions. As shown above the nearest

target address is selected. Since the BTAC contains predicted taken and not taken entries, either are selected depending on the corresponding offset and fetch address. If the corresponding instruction is a predicted taken branch then it is selected.

Claim Rejections - 35 USC § 103

47. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

> (a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

48. Claims 8-9, 17, and 37-38 are rejected under 35 U.S.C. 103(a) as being unpatentable over Shiell in view of Liu (6,088,793).

49. In regard to claim 8,

a. Shiell discloses the microprocessor of claim 1,

b. Shiell does not disclose wherein said BTAC is configured to provide a valid indicator associated with each of said plurality of target addresses for indicating whether said associated target address is a valid target address.

c. Liu has taught in column 4, lines 29-54 that a BTB (branch target buffer), or BTAC, provides a valid signal, or indicator, for each branch encountered in the cache (along with the branch target address) to the IFU (instruction fetch unit). Figure 6b shows that each entry of the BTB has a valid bit for this indication and the valid indication is associated with each address (branch tag and offset) of the entry.

d. The ability to know whether an address for branching is valid or not and therefore use only correct data, would have motivated one of ordinary skill in the art to modify the branch target address cache design of Shiell to use the valid indicator design of Liu.

It would have been obvious to one of ordinary skill in the art to modify the branch target address cache design of Shiell to incorporate the valid indicator taught by Liu so that only valid data is used for branching.

50. In regard to claim 9, Shiell in view of Liu discloses the microprocessor of claim 8, wherein said selector signal is used to select said one of said plurality of target addresses only if said associated valid indicator indicates that said target address is valid. Column 4, lines 29-54 of Liu show that if the BTB entry is valid then the branch address is used otherwise the next serial address is used.

51. In regard to claim 17,

a. Shiell discloses the apparatus of claim 15,

b. Shiell does not disclose wherein said control logic generates said selector to select said one of said plurality of target addresses having said smallest said corresponding one of said plurality of offsets if said BTAC indicates that said one of said plurality of target addresses is a valid target address.

c. Liu has taught in column 4, lines 29-54 that a BTB (branch target buffer), or BTAC, provides a valid signal, or indicator, for each branch encountered in the cache (along with the branch target address) to the IFU (instruction fetch unit). Figure 6b shows that each entry of the BTB has a valid bit for this indication and

the valid indication is associated with each address (branch tag and offset) of the entry.

d. The ability to know whether an address for branching is valid or not and therefore use only correct data, would have motivated one of ordinary skill in the art to modify the branch target address cache design of Shiell to use the valid indicator design of Liu.

It would have been obvious to one of ordinary skill in the art to modify the branch target address cache design of Shiell to incorporate the valid indicator taught by Liu so that only valid data is used for branching.

52. In regard to claim 37,

a. Shiell discloses the method of claim 29,

b. Shiell does not disclose the method further comprising: providing a plurality of indications of whether said corresponding plurality of target addresses is a valid target address.

c. Liu has taught in column 4, lines 29-54 that a BTB (branch target buffer), or BTAC, provides a valid signal, or indicator, for each branch encountered in the cache (along with the branch target address) to the IFU (instruction fetch unit). Figure 6b shows that each entry of the BTB has a valid bit for this indication and the valid indication is associated with each address (branch tag and offset) of the entry.

d. The ability to know whether an address for branching is valid or not and therefore use only correct data, would have motivated one of ordinary skill in the

art to modify the branch target address cache design of Shiell to use the valid indicator design of Liu.

It would have been obvious to one of ordinary skill in the art to modify the branch target address cache design of Shiell to incorporate the valid indicator taught by Liu so that only valid data is used for branching.

53. In regard to claim 38, Shiell in view of Liu discloses the method of claim 37, wherein said selecting comprises selecting as said second fetch address said one of said plurality of target addresses wherein a corresponding one of said plurality of indications indicates that said selected one of said plurality of target addresses is a valid target address. Column 4, lines 29-54 of Liu show that if the BTB entry is valid then the branch address is used otherwise the next serial address is used

Conclusion

54. The following is text cited from 37 CFR 1.111(c): In amending in reply to a rejection of claims in an application or patent under reexamination, the applicant or patent owner must clearly point out the patentable novelty which he or she thinks the claims present in view of the state of the art disclosed by the references cited or the objections made. The applicant or patent owner must also show how the amendments avoid such references or objections.

55. The prior art made of record and not relied upon is considered pertinent to applicant's disclosure. The following patents have been cited to further show the art with respect to branch target address caching in general.

US Pat No. 5,761,723 to Black shows a processor with branch speculation and a branch target address cache.

US Pat No 5,353,421 to Emma discloses processor with a combined branch history table and instruction fetching mechanism in one unit that functions as a branch target address cache.

US Pat No 6,502,185 to Keller teaches a processor including an indirect branch target cache.

US Pat No 5,812,839 to Hoyt gives a processor that uses a branch target buffer or cache for speculative execution.

Any inquiry concerning this communication or earlier communications from the examiner should be directed to Shane F Gerstl whose telephone number is (703)305-7305. The examiner can normally be reached on M-F 6:45-4:15 (First Friday Off).

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Eddie Chan can be reached on (703)305-9712. The fax phone number for the organization where this application or proceeding is assigned is 703-872-9306.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free).

Shane F Gerstl
Examiner
Art Unit 2183

SFG
March 15, 2004

EDDIE CHAN
SUPERVISORY PATENT EXAMINER
TECHNOLOGY CENTER 2100